Prototype Group Report

200 words summary

The EDR we choose for our project is to deploy a smart delivery system where users are able to generate an OTP and allows delivery personnel to unload the deliveries into house without the need to be physical available during the delivery process. In the beginning, we aim to accomplish the full detailed smart delivery management system but now we simplify to focus on main functionality which are the OTP generation and validation. For example, the implementing OTP generation through HOTP and the validating OTP with proper error handling and input validation. Additionally, the CLI interface is developed to simulate the process of OTP generation & validation and mimic the environment for testing to ensure the system functionality.

Furthermore, there are several software engineering techniques have been used to ensure a robust development process. For instance, version control, testing, build automation, dependency management and containerization via Docker. Continuous integration optimize the workflow by automatically testing and ensuring the code quality. In order to ensure security and consistency, OTPs are generated by using the HMAC-based OTP (HOTP) standard. Users can ensure the security and reliability in every step while performing all compulsory operations through the CLI. The accomplished prototype demonstrates the practicability of the EDRs design, providing a functional system and the strong foundation for future development.

800 WORDS GROUP REPORT- Project Management

We have distributed the prototype into 2 sections which are back-end development and CLI development. In the back-end development section, there will be 2 members focusing on developing the logics for OTP generation and OTP validation while on CLI development, the member will be focusing on building a command line interface that resembles the app and simulate the environment.

Work distributions:

Nonspecific assigned task:

1. Create a make file for building automation(work collaboratively)
2. Create continuous integration pipeline(work collaboratively)

Back-end Development (2 members):

Part 1: OTP Generation

Objectives: We aim to carry out the implementation into 2 phases with a core logic behind which is OTP generation. In phase 1, we create the OTP generation using HOTP method while phase 2 we aim to improve the system to TOTP method.

Deliverables:

1 code file for OTP generation with core functions listed below.

1 code file for the test case for testing each of the functions.

Work Updates:

Basic features:

Features	Status	Details
HOTP Algorithm Implementation	Completed	We use pyotp to import HOTP algorithm for generating the OTP.
Encryption & Decryption Mechanism	Completed	We have successfully implemented this features by using “cryptography.fernet”.
Add user function	Completed	Add user into the system before allowing them to generate an OTP(simulate to add a user and include into the system before using the features).
Delete user function	Completed	Delete an existing user(simulate where the user is not around and want remove him from the system).
OTP Generation Logic	Completed	The generate OTP method handles the logic behind for generating OTP.
OTP Validation Support	Completed	Support integrating with OTPvalidation class to crosscheck the OTP and performs validation.
Log records	Completed	Display all the logs history (mimic where user want to check back history).
Unit Test case	Completed	We set multiple fix cases to test functions in OTP generation.

Additional changes in Phase 2

Features	Status	Details
Generated Time Logging	Completed	We add the time log where the OTP is generated.
Validated Time Logging	Completed	We add the time log where the OTP is validated.
OTP status	Completed	We add status such as

		Validated, Not yet validated and Expired to the OTP.

Part 2: OTP Validation

Objectives: In OTP validation section, the member responsible for this role should produce the code for validating and write unit test to test individual functions are working.

Deliverables:

1 code file for OTP validation

1 code file for the test case

Work Updates:

Features	Status	Details
OTP Validation Function	Completed	The "OTP_validation" method handles the logic behind for generating OTP.
Match Check Function	Completed	Ensures the OTP matches the one stored/generated for validation.
Error handling	Completed	Implement logics for checking OTP format and used OTPs, printing appropriate error messages.
Unit Test case	Completed	We set multiple fix cases to test functions in OTP validation are working as expected.

CLI Development and Main program (1 member):

Objectives: Develop a main program that will integrate back-end logic for OTP generation and validation with a Command-Line Interface (CLI). The CLI simulates the application, including features such as adding/deleting users (to resemble people in the house), generating OTPs, and acting as a smart door system for entering and validating OTPs, providing an environment for us to test our system.

Deliverables:

1 code file for CLI

1 code file for main program.

Work Updates:

Features	Status	Details
Build CLI Interface	Completed	Created a CLI for OTP system interaction.
Simulate add user function	Completed	Add a user to the system

Simulate delete user function	Completed	Delete a user to the system
Simulate OTP generation	Completed	Generate OTP for a specific user.
Simulate OTP validation	Completed	Validate OTP entered by the user in a smart door simulation.
Simulate show user	Completed	Show list of user available in the system
Display log	Completed	Display log records

Additional changes in Phase 2

Features	Status	Details
Add colors to CLI	Completed	Enhanced the CLI with color-coded outputs for better user visibility

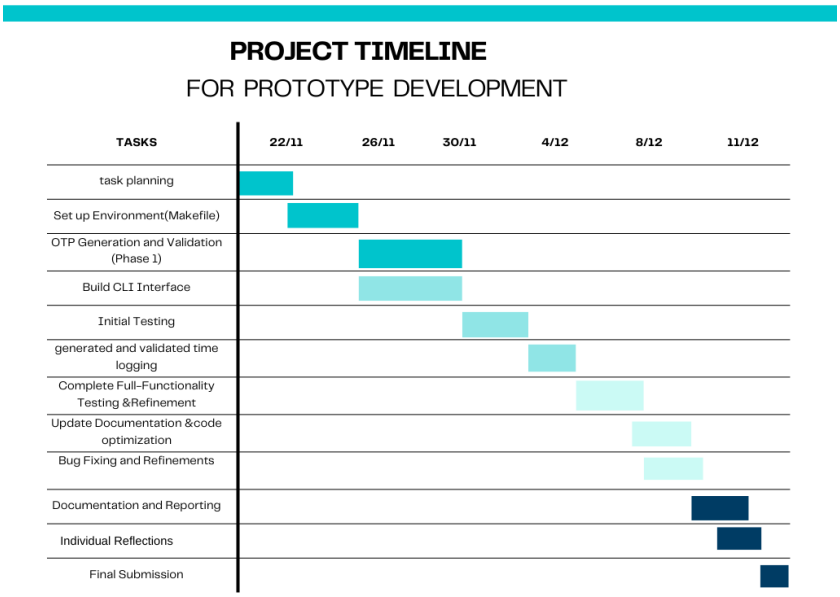
Integration Testing:

After completing each individual parts, we combined all the individual components and integrated into Main.py and carry out integration testing to ensure out functions all work seamlessly together.

Key changes in our plans:

We have decided not to continue with TOTP method for our phase 2 due to unfit alignment with our idea. We aim to provide convenience to user by generate an OTP for the delivery personnel and no longer need to worry about it due to the OTP will remain valid until the delivery is made however, if we implemented TOTP, the OTP will be expired before the delivery is made leading to defeating the purpose of this system. Instead, we have add additional functions such as add OTP expiration status, show time generated and validated of the OTP.

Work Timeline:



Acknowledgement of AI:

We have identified the use of generative AI to generate our test data output. However each test cases ideas are our own ideas to test the possible outcomes based on our code.

Peer reflection by member 1(Colin):

Reflections for Keng Xiang:

Overall, Keng Xiang is collaborator when working with the team. He has good communication and listening skills hence, allowing him to handle any miscommunications that happens between the team. When discrepancy of ideas occur during the team, Keng Xiang will tend to listen to both side, and analysis the situation and provide constructive feedbacks.

Furthermore, he has demonstrated good collaborative skills when working together with me, particularly on the OTP validation part when both parts are interconnected. When we are working on these part, Keng Xiang has constantly approach me to ensure both are us are working towards the same goals instead of out of track. Whenever there's enquiry or susceptions towards a problem, he will approach and discuss with me regarding the issues and seek for solutions together if there's an error.

Keng Xiang is also an independent and resourceful when it comes to solving problems. He tends to seek for answers that can solve his enquiries himself before asking for help and furthermore, he always go beyond and will try to search for better solutions.

However, 1 key improvement for Keng Xiang is he need to work on time management. Often, due to his independent approach, he tend to waste too much on some non-crucial task leading delays for upcoming task. Hence, I will recommend he evaluate the importance of each task then assign them to with a suitable time frame so he won't allocate too much time and hinder the process of the work.

Reflections for Chenyu(Max):

Max has demonstrated his ability to follow instructions and work together as a team. He is able to complete the work assign to him within the lime allocated and will constantly keep the other members updates and enquire if there are any updates needed. This has made him to be a good cooperative team member however there are a few improvements he can work on to be better.

On contrary to Keng Xiang, Max tend to rely on other people when he needed help on a specific part instead of working independently first to seek for solution. This might has cause trouble to other members as they required to stop their working process and assist him which eventually cause to delay of other works. Hence, I would recommend him to try to approach a solution independently first and only ask for assistance if a solution could not be solved individually.

Besides, I would also recommend Max to be more engaging in team discussion instead of being a passive listener. It's good where he follow instructions and carry out the task successfully, however, in a team collaborative environment, brainstorming ideas and sharing personal insights could help to identify certain ideas that is good but being missed out by the rest of the members.

Peer reflection by member 2(Keng Xiang):

Reflection for Colin:

Teammate Colin is our group leader, who plays an important role in our project. His task is about

OTP generation and creating the Makefile. I think this part is difficult and challenging However, he done a good job on it because of his technical proficiency and good communication skills as a leader. For example, his inclusion of the encrypt_OTPsecret method was a crucial addition to our project. It added an essential layer of security by encrypting sensitive OTP secrets before storage. This implementation demonstrated his strong understanding of cryptography and secure coding practices.

Additionally, Colin also created the Makefile, which automated repetitive tasks such as testing and running the application. This helps us ensuring consistency and reduced setup time for all members. One area for improvement could be to allocate more time for testing and troubleshooting. Teammate A did an excellent job with the OTP generation and Makefile.

Furthermore, Colin also proficient in management of the CI pipeline. He developed a robust CI process which allow the system can be continuous testing and integration of code changes.

Personally, I gained the basic CI techniques knowledge from Colin. Although the are of CI is my weakness, he still share insights and provide guidance enhanced my understanding of automated testing

However, I think we must dedicating additional time to test the implementation. This extra time for testing could allowed us to refine the overall functionality, ensuring that all components worked well. If any issues came up, we would have had more time to fix them and improve the overall efficiency of the project.

Reflection for Chenyu(Max):

Chenyu is responsible for developing the CLI. However, he faced some technical expertise when complete the task. For example, the building of CLI and the integration test. So, this might lead to some work delay. Luckily, He has positive attitude and always open to feedback and assistance. Throughout the project, Colin and I always provided guidance and support to Chenyu, such as integrating the CLI with the OTP generation and validation modules and handling user input validation. He also provide his suggestion for our work. In our group, we supported one another, which allowed our work to progress smoothly.

One improvement that I think would be task planning at the start of the project. We might have clearer discussions about each team member's strengths and weakness. This could helped us assign tasks more effectively and avoid delays. Also, I think that regular progress check could also have made it easier to solve issues quickly. Another improvement would be for Chenyu to improve his technical skills especially in programming and debugging. This would not only help us complete our work more efficiently but also contribute to his personal growth in his other projects. I believe that improving these areas will help us work even better together in future projects.

Peer reflection by member 3(Max):

Reflection for Colin:

Strengths and Contributions:

Colin is an excellent colleague with excellent programming skills. His work on the design and implementation of OTP generation logic was critical to the success of our project.

Whenever we face challenges, Colin skilfully guides us through them with clear insights and effective solutions.

At the same time, he actively shares knowledge, discusses bug fixes, and recommends valuable

books. This willingness to support others creates a productive and collaborative work environment. His reliability and dedication have made him a cornerstone of our team.

Suggestions for Future Collaboration:

As we continue working together beyond this project, I would suggest that Colin balance Perfection with Deadlines: Setting realistic time-bound goals while maintaining his high standards would streamline progress and help meet deadlines more efficiently.

Conclusion:

Colin's leadership has been a great help to our group assignments and I look forward to continuing the good working relationship.

Reflection for Keng:

Strengths and Contributions:

Keng is a dedicated and responsible colleague whose work on OTP validation and error handling is very precise, ensuring that the code works. Keng's attention to detail and ability to handle edge situations added critical reliability to the project.

His proactive communication ensures that everyone is on the same page with progress and expectations, creating a good working environment for everyone.

Suggestions for Future Collaboration:

In order to make our future cooperation more effective, I suggest that we make a schedule for our group discussions. Although he always brings new ideas to the discussion, I admit that I am occasionally late and adjusting the schedule can improve the efficiency of our meetings.

Closing Notes:

Both Colin and Keng have demonstrated exceptional qualities that make them invaluable colleagues. I look forward to working with them on future projects and believe that the suggestions provided will help further enhance our collaborative efforts in a professional setting.